IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Appellant(s): Surya Varanasi, Timothy Millet	§	Art Unit: 2416
	§	Confirmation No.: 1584
Serial No.: 10/699,567	§	Examiner: Chandrahas B. Patel
Filed: October 31, 2003	§	Docket No. 112-0134US
For: Logical Ports in Trunking	§	Customer No.: 85197

APPEAL BRIEF

Via USPTO EFS **March 2, 2009**
Commissioner for Patents
P. O. Box 1450
Alexandria, VA 22313-1450
Mail Stop: Appeal Briefs – Patents

Appellants hereby submit this Appeal Brief in connection with the above-identified application. A Notice of Appeal was filed on January 2, 2009.

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST 3
II. RELATED APPEALS AND INTERFERENCES 4
III. STATUS OF CLAIMS 5
IV. STATUS OF AMENDMENTS 6
V. SUMMARY OF CLAIMED SUBJECT MATTER 7
VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL 13
VII. ARGUMENT 14
A. Overview of Yamada 14
B. Description of Munter 15
1. Overview 15
2. The Packet Distribution Taught by Munter 17
3. The Load Balancing Taught by Munter 19
C. The Obviousness Rejections 20
4. Independent claims 1, 21, 41, 61, 81 and 101 20
a. The Load Balancing of the Independent Claims 22
b. Munter does not Teach Balancing across Trunked Ports 23
c. Munter does not Teach Balancing across Logical Ports within a Switch that Includes the Trunked Group Ports 25
d. Munter does not Teach Two-Tied Balancing within a Single Switch 26
5. Dependent claims 2, 6-20, 22, 26-40, 42, 46-60, 66-80, 82, 86-100 and 103-105 28
D. Conclusion 29
VIII. CLAIMS APPENDIX 31
IX. EVIDENCE APPENDIX 47
X. RELATED PROCEEDINGS APPENDIX 48

I. REAL PARTY IN INTEREST

The real party in interest is Brocade Communications Systems, Inc.

II. RELATED APPEALS AND INTERFERENCES

Appellants are unaware of any related appeals or interferences.

III. STATUS OF CLAIMS

Originally filed claims:	1-105.
Added claims:	None.
Claim cancellations:	3-5, 23-25, 43-45, 63-65, 83-85 and 102.
Presently pending claims:	1-2, 6-22, 26-42, 46-62, 66-82, 86-101 and 103-105.
Presently appealed claims:	1-2, 6-22, 26-42, 46-62, 66-82, 86-101 and 103-105.
Presently allowed claims:	None.
Presently objected claims:	None.

IV. STATUS OF AMENDMENTS

There were no amendments filed subsequent to the Office Action of September 22, 2008 (hereinafter "Final Office Action").

V. SUMMARY OF CLAIMED SUBJECT MATTER

This section provides a concise explanation of the subject matter defined in each of the independent claims involved in the appeal. Each element of the claims is identified with a corresponding reference to the specification and drawings where applicable. Note that the citation to passages in the specification and drawings for each claim element does not imply that the limitations from the specification and drawings should be read into the corresponding claim element.

Embodiments according to the presently claimed invention provide for the use of logical ports in the trunking of communication links between network switches. Such trunking allows frames to take a variety of alternate paths and still be delivered in-order at the desired destination.[1] A collection of links between two switches form a trunked group,[2] and frames are transmitted from one switch to the other by distributing frames across the links (and their associated switch ports) while preserving in-order delivery of the frames.[3] A correspondence is employed between logical ports and physical ports of a switch, wherein a logical port refers to a collection of one or more physical ports.[4] When a port is selected for a frame exiting a switch over a trunked group that includes one or more logical ports, the selection of the physical port is based at least in part on the correspondence between the logical ports and the physical ports.[5] If a physical port is itself associated with a link that is part of a trunked group, all of the links (and associated ports) within the trunked group are treated as a single "logical pipe,"[6] wherein frames are distributed across the links (and ports) within the trunked group.[7]

In accordance with the invention of independent claim 1, for example, a method of routing a flow of frames is claimed that includes applying a correspondence between a plurality

---

[1] Specification of the subject Application as published (hereinafter "Specification"), lines 1-7 of ¶ [0025].
[2] Specification, lines 3-6 of ¶ [0026]; Fig. 3, trunked group 300.
[3] Specification, lines 6-15 of ¶ [0027].
[4] Specification, lines 4-8 of ¶ [0030].
[5] Specification, lines 7-11 of ¶ [0031].
[6] Specification, lines 4-6 of ¶ [0027]; Fig. 3, trunked group 300.
[7] Specification, lines 6-11 of ¶ [0027].

of logical ports and a plurality of physical ports of a switch,[8] at least one logical port having corresponded a plurality of physical ports to form a trunked group.[9] The corresponded physical ports can be any of the plurality of physical ports exiting the switch,[10] wherein the corresponded ports all operate at the same rate,[11] and wherein frames in a trunked group are delivered in order.[12] The method further includes balancing frame traffic through the switch using the plurality of logical ports,[13] the frame traffic including frames exiting the switch via the physical ports.[14] A selected physical port for at least one of the frames exiting the switch is selected based at least in part on the correspondence,[15] with any frames exiting the switch via physical ports forming a trunked group being balanced over the physical ports forming the trunked group.[16]

In accordance with the invention of independent claim 21, for example, an apparatus is claimed, including a switch that includes a processor and memory.[17] The switch further includes a plurality of logical[18] and a plurality of physical ports,[19] and has the capability to route a flow of frames exiting the switch.[20] The switch is adapted to apply a correspondence between the plurality of logical ports and the plurality of physical ports of the switch,[21] at least one logical port having corresponded a plurality of physical ports to form a trunked group.[22] The corresponded physical ports can be any of the plurality of physical ports exiting the switch.[23]

---

[8] Specification, lines 1-6 of ¶ [0030].
[9] Specification, lines 15-26 of ¶ [0031]; Fig. 3, ports 306-321 and trunked group 300.
[10] Specification, lines 6-9 of ¶ [0030].
[11] As noted in Appellants' prior Response of June 17, 2008, a prerequisite to trunking a port is that the port must operate at the same rate as the trunk master of interest (*i.e.*, it is an inherent characteristic of a trunked group that its constituent links all operate at the same rate). *See* patent application 09/872,412 (incorporated by reference into the subject Application), p. 26, lines 2-5 of ¶ [0065] and p. 27, lines 1-2 of ¶ [0067].
[12] Specification, lines 1-6 of ¶ [0030].
[13] Specification, lines 11-15 of ¶ [0031].
[14] Specification, lines 5-7 of ¶ [0031].
[15] Specification, lines 7-11 of ¶ [0031].
[16] Specification, lines 4-11 of ¶ [0027]; Fig. 3, trunked group 300.
[17] Specification, lines 1-9 of ¶ [0028]; Fig. 3, processors 330 and 331, and memory 334 and 335.
[18] Specification, lines 1-9 of ¶ [0030].
[19] Specification, lines 15-17 of ¶ [0027]; Fig. 3, ports 306-321; and lines 1-5 of ¶ [0029].
[20] Specification, lines 1-6 of ¶ [0043].
[21] Specification, lines 1-6 of ¶ [0030].
[22] Specification, lines 15-26 of ¶ [0031]; Fig. 3, ports 306-321 and trunked group 300.
[23] Specification, lines 1-9 of ¶ [0030].

The corresponded ports all operate at the same rate,[24] and frames in a trunked group are delivered in order.[25] The switch is further adapted to balance frame traffic through the switch using the plurality of logical ports,[26] the frame traffic including frames exiting the switch via the physical ports.[27] A selected physical port for at least one of the frames exiting the switch is selected based at least in part on the correspondence,[28] with any frames exiting the switch via physical ports forming a trunked group being balanced over the physical ports forming the trunked group.[29]

In accordance with the invention of independent claim 41, for example, a switch fabric is claimed,[30] including at least a first switch and a second switch.[31] The first switch includes a processor and memory,[32] as well as a plurality of logical ports[33] and a plurality of physical ports,[34] and has the capability to route a flow of frames exiting the first switch.[35] The first switch is adapted to apply a correspondence between the plurality of logical ports and the plurality of physical ports,[36] at least one logical port having corresponded a plurality of physical ports to form a trunked group.[37] The corresponded physical ports can be any of the plurality of physical ports exiting the switch,[38] wherein the corresponded ports all operate at the same rate,[39] and wherein frames in a trunked group are delivered in order.[40] The first switch is further adapted to balance frame traffic through the first switch using the plurality of logical ports,[41] the frame

---

[24] It is an inherent characteristic of a trunked group that its constituent links all operate at the same rate; *see* note 11.
[25] Specification, lines 1-6 of ¶ [0030].
[26] Specification, lines 11-15 of ¶ [0031].
[27] Specification, lines 5-7 of ¶ [0031].
[28] Specification, lines 7-11 of ¶ [0031].
[29] Specification, lines 4-11 of ¶ [0027]; Fig. 3, trunked group 300.
[30] Specification, lines 1-3 of ¶ [0021] and Fig. 1, fabric 110.
[31] Specification lines 1-4 and 15-19 of ¶ [0023]; and lines 1-5 of ¶ [0026] and Fig. 3, switches 302 and 304.
[32] Specification, lines 1-9 of ¶ [0028]; Fig. 3, processors 330 and 331, and memory 334 and 335.
[33] Specification, lines 1-9 of ¶ [0030].
[34] Specification, lines 15-17 of ¶ [0027]; Fig. 3, ports 306-321; and lines 1-5 of ¶ [0029].
[35] Specification, lines 1-6 of ¶ [0043].
[36] Specification, lines 1-6 of ¶ [0030].
[37] Specification, lines 15-26 of ¶ [0031]; Fig. 3, ports 306-321 and trunked group 300.
[38] Specification, lines 1-9 of ¶ [0030].
[39] It is an inherent characteristic of a trunked group that its constituent links all operate at the same rate; *see* note 11.
[40] Specification, lines 1-6 of ¶ [0030].
[41] Specification, lines 11-15 of ¶ [0031].

traffic including frames exiting the first switch via the physical ports.[42] A selected physical port for at least one of the frames exiting the first switch is selected based at least in part on the correspondence,[43] with any frames exiting the switch via physical ports forming a trunked group being balanced over the physical ports forming the trunked group.[44]

In accordance with the invention of independent claim 61, for example, a network is claimed,[45] including a host, a physical storage unit,[46] and a first switch and a second switch[47] communicatively coupled to form a switch fabric.[48] The first switch and the second switch are further communicatively coupled to the host and the physical storage unit.[49] The first switch includes a processor and memory,[50] as well as a plurality of logical ports[51] and a plurality of physical ports.[52] The first switch is adapted to apply a correspondence between the plurality of logical ports and the plurality of physical ports,[53] at least one logical port having corresponded a plurality of physical ports to form a trunked group.[54] The corresponded physical ports can be any of the plurality of physical ports exiting the switch,[55] wherein the corresponded ports all operate at the same rate,[56] and wherein frames in a trunked group are delivered in order.[57] The first switch is further adapted to balance frame traffic through the first switch using the plurality of logical ports,[58] the frame traffic including frames exiting the first switch via the physical ports.[59] A selected physical port for at least one of the frames exiting the first switch is selected

---

[42] Specification, lines 5-7 of ¶ [0031].
[43] Specification, lines 7-11 of ¶ [0031].
[44] Specification, lines 4-11 of ¶ [0027]; Fig. 3, trunked group 300.
[45] Specification, lines 1-3 of ¶ [0019]; Fig. 1, communications network system 100.
[46] Specification, lines 1-13 of ¶ [0020]; Fig. 1, devices 120, 122 and 124.
[47] Specification lines 1-4 and 15-19 of ¶ [0023]; and lines 1-5 of ¶ [0026] and Fig. 3, switches 302 and 304.
[48] Specification, lines 1-3 of ¶ [0021] and Fig. 1, fabric 110.
[49] Specification, lines 1-13 of ¶ [0020]; Fig. 1, devices 120, 122 and 124; lines 1-3 of ¶ [0021] and Fig. 1, fabric 110; lines 1-4 and 15-19 of ¶ [0023]; and lines 1-5 of ¶ [0026] and Fig. 3, switches 302 and 304.
[50] Specification, lines 1-9 of ¶ [0028]; Fig. 3, processors 330 and 331, and memory 334 and 335.
[51] Specification, lines 1-9 of ¶ [0030].
[52] Specification, lines 15-17 of ¶ [0027]; Fig. 3, ports 306-321; and lines 1-5 of ¶ [0029].
[53] Specification, lines 11-15 of ¶ [0031].
[54] Specification, lines 15-26 of ¶ [0031]; Fig. 3, ports 306-321 and trunked group 300.
[55] Specification, lines 1-9 of ¶ [0030].
[56] It is an inherent characteristic of a trunked group that its constituent links all operate at the same rate; *see* note 11.
[57] Specification, lines 1-6 of ¶ [0030].
[58] Specification, lines 11-15 of ¶ [0031].
[59] Specification, lines 5-7 of ¶ [0031].

based at least in part on the correspondence,[60] with any frames exiting the switch via physical ports forming a trunked group being balanced over the physical ports forming the trunked group.[61]

In accordance with the invention of independent claim 81, for example, an article is claimed including a storage medium having stored thereon instructions, that when executed, result in performance of a method of routing a flow of frames.[62] The method performed includes applying a correspondence between a plurality of logical ports and a plurality of physical ports of a switch,[63] at least one logical port having corresponded a plurality of physical ports to form a trunked group.[64] The corresponded physical ports can be any of the plurality of physical ports exiting the switch,[65] wherein the corresponded ports all operate at the same rate,[66] and wherein frames in a trunked group are delivered in order.[67] The method performed further includes balancing frame traffic through the switch using the plurality of logical ports,[68] the frame traffic including frames exiting the switch via the physical ports.[69] A selected physical port for at least one of the frames exiting the switch is selected based at least in part on the correspondence,[70] with any frames exiting the switch via physical ports forming a trunked group being balanced over the physical ports forming the trunked group.[71]

In accordance with the invention of independent claim 101, for example, an article is claimed, including a storage medium having stored thereon instructions that, when executed, result in performance of a method of initializing a switch to route a flow of frames.[72] The method performed includes initializing a correspondence between a plurality of logical ports and

---

[60] Specification, lines 7-11 of ¶ [0031].
[61] Specification, lines 4-11 of ¶ [0027]; Fig. 3, trunked group 300.
[62] Specification, lines 1-23 of ¶ [0057].
[63] Specification, lines 1-6 of ¶ [0030].
[64] Specification, lines 15-26 of ¶ [0031]; Fig. 3, ports 306-321 and trunked group 300.
[65] Specification, lines 6-9 of ¶ [0030].
[66] It is an inherent characteristic of a trunked group that its constituent links all operate at the same rate; *see* note 11.
[67] Specification, lines 1-6 of ¶ [0030].
[68] Specification, lines 11-15 of ¶ [0031].
[69] Specification, lines 5-7 of ¶ [0031].
[70] Specification, lines 7-11 of ¶ [0031].
[71] Specification, lines 4-11 of ¶ [0027]; Fig. 3, trunked group 300.
[72] Specification, lines 1-23 of ¶ [0057].

a plurality of physical ports of the switch,[73] at least one logical port having corresponded a plurality of physical ports to form a trunked group.[74] The corresponded physical ports can be any of the plurality of physical ports exiting the switch,[75] wherein the corresponded ports all operate at the same rate,[76] and wherein frames in a trunked group are delivered in order.[77] The method further includes initializing the switch to balance frame traffic[78] through the switch using the plurality of logical ports,[79] the frame traffic including frames exiting the switch via the physical ports.[80] A selected physical port for at least one of the frames exiting the switch is selected based at least in part on the correspondence,[81] with any frames exiting the switch via physical ports forming a trunked group being balanced over the physical ports forming the trunked group.[82]

---

[73] Specification, lines 10-12 of ¶ [0057].
[74] Specification, lines 15-26 of ¶ [0031]; Fig. 3, ports 306-321 and trunked group 300.
[75] Specification, lines 6-9 of ¶ [0030].
[76] It is an inherent characteristic of a trunked group that its constituent links all operate at the same rate; *see* note 11.
[77] Specification, lines 1-6 of ¶ [0030].
[78] Specification, lines 12-13 of ¶ [0057].
[79] Specification, lines 11-15 of ¶ [0031].
[80] Specification, lines 5-7 of ¶ [0031].
[81] Specification, lines 7-11 of ¶ [0031].
[82] Specification, lines 4-11 of ¶ [0027]; Fig. 3, trunked group 300.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

- Whether claims 1-2, 6-22, 26-42, 46-62, 66-82, 86-101 and 103-105 are obvious over Yamada et al. (U.S. Pat. No. 7,203,762, hereinafter "Yamada") in view of Munter (U.S. Pat. No. 7,209,659, hereinafter "Munter"), and further in view of Battle et al. (U.S. Pat. No. 7,088,713).

VII. ARGUMENT

The claims do not stand or fall together. Instead, Appellants present separate arguments for various independent and dependent claims. After a concise discussion of cited art, each of these arguments is separately argued below and presented with separate headings and sub-heading as required by 37 CFR § 41.37(c)(l)(vii).

A. Overview of Yamada

Yamada is directed to a system that allows both network layer (layer-3) and data link layer (layer-2) virtual private networks (VPNS) to be implemented on a single physical network.[83] A multi-protocol label-switch virtual private network (MPLS-VPN) is provided for transport of MPLS frames,[84] wherein layer-2 and layer-3 messages are encapsulated by MPLS frames that each includes a label[85] added to identify the layer to which the encapsulated message belongs.[86] Tables are maintained that are used to route frames received by an edge node.[87] These tables map the layer destination addresses to particular virtual sending ports, and the virtual sending ports to particular MPLS-side physical ports.[88] Based upon the identified MPLS-side physical port, a label is added to the encapsulating MPLS frame, and the frame is transmitted to the MPLS network through the identified MPLS-side physical port.[89]

Thus Yamada teaches receiving a frame, determining the destination address of the received frame, determining a single corresponding MPLS-side physical port to be used for transmission of the frame, adding a label based upon the mapped physical port, and transmitting the frame over the mapped physical port. Appellants note that the selection of an MPLS-side physical port used for transmission of the frame is based upon a one-to-one mapping from a virtual sending port to a single MPLS-side physical port, as shown by the L1 mapping table T6

[83] Yamada, Abstract.
[84] Yamada, col. 2, lines 52-58.
[85] Yamada, col. 2, line 58.
[86] Yamada, col. 3, lines 1-8.
[87] Yamada, col. 4, lines 56-59.
[88] Yamada, col. 7 line 51 through col. 8, line 8.
[89] Yamada, col. 8, lines 8-13.

shown in Fig. 6 of Yamada. Appellants further note that Yamada does not teach or suggest more than one MPLS-side physical port selection for transmission of a given frame.

B. Description of Munter

1. Overview

Munter is directed to a networking system that allows for increasing its capacity by incrementally adding planes of switches to core nodes.[90] The core network uses optical switching with wavelength division multiplexing (WDM), wherein each plane of core switches is a virtual parallel plane (VPP) network that is assigned to a particular optical wavelength.[91] Traffic from other networks enters the core network (within what is referred to as the "backbone" layer) through several edge nodes (within what is referred to as the "point-of-presence" or "POP" layer) that operate to funnel data packets into the core network.[92] Each edge node (also referred to as a POP switch[93]) connects to each core node within the core network via an edge trunk consisting of a number of parallel serial links (*e.g.*, multiple 10 Gbs SONET links wavelength division multiplexed on one or more physical fibers).[94]

Each serial link within an edge trunk is associated with a wavelength, and thus with a port on a core switch.[95] Traffic associated with each individual serial link within an edge trunk (*e.g.*, serial link R1-1) is routed by an optical cross connect or OCX (also referred to as an optical switch, and as an optical distribution unit or ODU[96]) between a distribution point associated with the edge node (*e.g.*, distribution point K1 associated with edge node R1) and a corresponding core switch port link (*e.g.*, core switch port link S1R1).[97] The function of the optical cross connect is to connect the appropriate optical wavelengths (*i.e.*, colors) to the corresponding nodes by splitting the wavelengths of a optical signal received from a fiber, switching each

---

[90] Munter, Abstract.
[91] *Id.*
[92] Munter, col. 3, lines 26-32 and lines 38-41; and Fig. 1, POP switches R1 through R4.
[93] Munter, col. 3, lines 30-31.
[94] Munter, col. 3, lines 33-38; and Fig. 2 (*e.g.*, trunk 25 and serial links R1-1, R1-2, R1-3 and R1-4).
[95] Munter, col. 3, lines 65-67; and Fig. 2 (*e.g.*, trunk 25 and serial links R1-1, R1-2, R1-3 and R1-4).
[96] Munter, col. 7, lines 46-49; and Fig. 5, ODU (OCX) 60.
[97] Munter, col. 5, lines 18-28; and Fig. 4, OCX 40.

individual color signal, and recombining individual color signals to form a new optical signal for output.[98]

Each core switch port link may be either an individual optically switchable S-color connection,[99] or alternatively a "bundle" of multiple wavelength links (*i.e.*, a collection of individual optically switchable S-color connections) that together operate as a single logical port link.[100] In an individual-connection port link configuration, the number of core switch ports used to connect to the POP switches is equal to the number of POP switches[101] (*i.e.*, one port per port link). In a bundled-connection port link configuration, the number of core switch ports used to connect to the POP switches is equal to an integer multiple m of the number of POP switches[102] (thus allowing, *e.g.*, m ports per port link). For non-symmetrical configurations, *e.g.*, where corresponding switch port links associated with different switches (*e.g.*, S1R1, S2R1, S3R1 and S4R1) have different capacities (*e.g.*, some are bundled, some are not), a load balancing algorithm takes into account the imbalance[103] (*i.e.*, takes into account differences in capacities of links in different parallel backbone networks[104]).

Core nodes are similarly connected to each other by core links that also consist of any number of parallel links (*e.g.*, 10 Gbs SONET links wavelength division multiplexed on one or more physical fibers).[105] Each serial link within a core link is associated with a wavelength, and thus with a port on a core switch.[106] Traffic associated with each individual serial link (*i.e.*, each individual optically switchable S-color link[107]) within a core link (*e.g.*, serial link S1a) is routed by an optical cross connect or OCX between a distribution point associated with another core

---

[98] Munter, col. 9, lines 31-35.
[99] Munter, col. 5, lines 25-28; and Fig. 4, individual optically switchable S-color connections S1R1, S1R2, S1R3 and S1R4.
[100] Munter, col. 5, lines 29-31.
[101] Munter, col. 5, lines 25-28; and Fig. 4, individual optically switchable S-color connections S1R1, S1R2, S1R3 and S1R4.
[102] Munter, col. 5, lines 20-22.
[103] Munter, col. 5, lines 31-33.
[104] Munter, col. 6, lines 29-32.
[105] Munter, col. 3, lines 33-38; and Fig. 2 (*e.g.*, trunk 25 and serial links R1-1, R1-2, R1-3 and R1-4).
[106] Munter, col. 4, lines 21-30; and Fig. 3 (*e.g.*, trunk 12 and serial link S1a (and by logical implication serial links S2a, S3a, S4a and S5a)).
[107] Munter, col. 4, lines 37-43.

node (*e.g.*, distribution point Ta associated with core node A) and a distribution point associated with a corresponding core link (*e.g.*, distribution point Pa associated with core link S1a).[108]

Each core link may be expanded by adding wavelengths to the core link (resulting in multiple wavelengths being associated with a core link) and adding ports to each of the core switches (one port per added wavelength at each switch).[109] However, because traffic is split evenly and indiscriminately at each POP switch to access all of the parallel backbone networks equally, any striping algorithm implemented at the POP switch to balance traffic must take into account non-symmetrical configurations, *e.g.*, wherein different VPP networks provide different link and node capacities.[110] Such traffic balancing is accomplished through the use of a non-synchronous rotation scheme similar to what is frequently described as "striping," wherein all packets sent from an edge node towards the core nodes are systematically distributed over all links.[111]

Munter thus teaches balancing data traffic across parallel backbone networks through the use of a striping algorithm that evenly splits traffic at the POP switch across all of the backbone networks by distributing the traffic over all links between the POP switches and the core nodes of the network, and that adjusts the striping to take into account any capacity asymmetries between the parallel backbone networks.

2. The Packet Distribution Taught by Munter

Appellants note that with regard to distributing packets from an edge node to a core node, Munter teaches that,

> FIG. 4 illustrates core node D having five VPP networks connected to four POP switches R1 to R4 through trunks 26-29. To connect POP switches R1 to R4 to the core node D, an optical cross connect (OXC) 40 is used on the access side of node D. Modular switches S1 to S5 should have four, or multiple of four access ports, to connect to four POP switches R1 to R4 through bi-directional trunks 26-29, **while distribution points K1 to K4 on OXC 40 should be each**

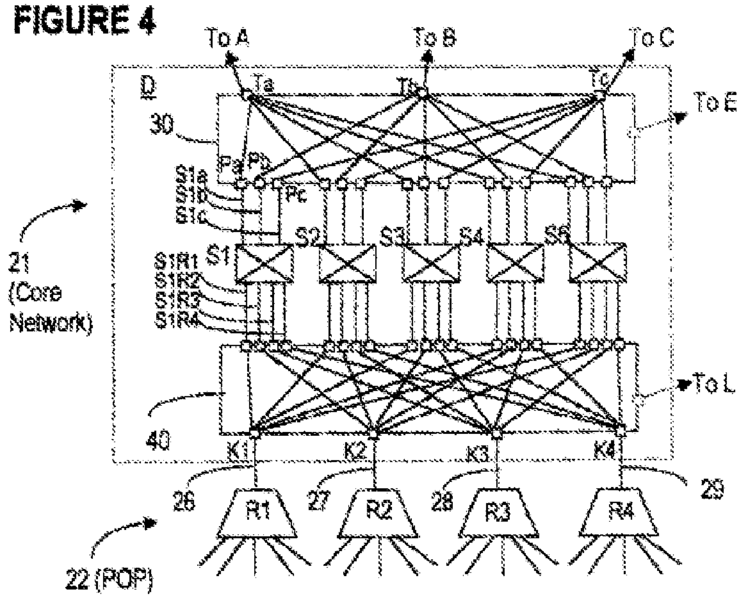
---

[108] Munter, col. 4, lines 30-37; and Fig. 4, OCX 30.
[109] Munter, col. 4, lines 50-55 and Fig. 3.
[110] Munter, col. 6, lines 24-31.
[111] Munter, col. 9, lines 60-63.

> **provided with five bi-directional links to switches [S]1 to S5**. For example, **switch S1 is connected to POP switches R1 to R4 through four bi-directional port links**, or individual optically switchable S-color connections **S1R1, S1R2, S1R3, and S1R4**.
>
> **Each logical port link S1R1** practically includes a "bundle" of wavelength links which in fact is a bundle of ports. **The links** may have different capacity **as long as the un-symmetry, or the imbalance is taken into account**. This is done by **a load balancing algorithm**, as known in the art.[112]

The cited text from Munter above focuses on how the POP switches (also referred to as edge nodes[113]) connect to the core nodes, and more specifically on how the optical cross connect provides frames to and from specific POP switches over dedicated individual bi-directional port links that each connects to a core switch associated with a specific backbone network plane. Traffic to and from each POP switch is distributed across the various planes of the backbone network by distributing the traffic across switches S1, S2, S3, S4 and S5 over such port links.[114] As can be seen in Fig. 4 of Munter (reproduced at left), traffic from POP switch R1 is split into five data streams by optical cross connect 40, and each stream is routed to a specific switch across one of five port links, each connected to a different switch. Thus, *e.g.*, the portion of POP switch R1's traffic that is routed to and from POP switch S1 traverses port link S1R1. The remainder of the traffic is similarly distributed across the remaining links (referred to for purposes of the discussion as S2R1, S3R1,

FIGURE 4



[112] Munter, col. 5, lines 17-34 (emphasis added).
[113] *See* Munter, col. 3, lines 38-41.
[114] *See* Munter, col. 6, lines 24-29. Appellants note that although Munter describes distributing traffic that originates from a POP switch, by implication traffic to a POP switch is also so distributed, since the received traffic originates from another POP switch.

S4R1 and S5R1) that connect POP switch R1 (via cross connect 40) to each of the remaining switches S2-S5. If the port links have different capacities, thus resulting in a capacity imbalance between parallel planes within a node, the load balancing taught by Munter is designed to take the imbalance into account.

3. The Load Balancing Taught by Munter

Appellants further note that the load balancing referenced throughout Munter is taught within the context of distributing traffic across the virtual parallel planes.[115] Indeed, one of the objectives of the parallel plane structure of the core nodes taught by Munter is to facilitate expanding the capacity of the core network by adding additional parallel planes over which traffic is then distributed.[116] This distribution is accomplished in Munter by implementing a striping algorithm at the POP switches which distributes traffic across the available parallel planes (*i.e.*, across core switches within a core node[117]) as the packets enter the network. Specifically, Munter teaches that,

> The network topology includes N parallel backbone networks of equal, or different capacities. Each POP switch Ri has equal access to all N parallel backbone networks, but different access values may be also considered. **Routing across <u>each parallel backbone</u> network operates <u>equally</u>**, i.e. each core node A to D is considered a single router, and the forwarding tables in each of the modular switches Si that make up a core node A to D are identical. **Traffic is split evenly and nondiscriminatory <u>at each POP switch</u> Ri to <u>access all N parallel backbone networks equally</u>, for example based on a striping algorithm designed <u>to balance the traffic</u> while preserving flow packet order.** The striping algorithm needs **to <u>take into account</u> the actual link capacities** in the case where full symmetry is not required, e g. different link and node capacities in different VPP networks.[118]

---

[115] *See*, *e.g.*, Munter, col. 6, lines 24-29 (describing how the POP switches distribute traffic); col. 8, lines 50-53 (describing how the striping algorithm balances traffic over the core nodes); col. 9, lines 61-63 (describing how all packets sent by the edge nodes towards the core nodes are distributed over all links); and col. 11, lines 6-9 (describing how different capacity nodes and links can be mixed as long as each edge node has access to all core nodes and the node capacities are factored into the striping algorithm).
[116] *See* Munter, col. 2, lines 39-42 ("The present invention allows linear scaling of the existing network through the creation of virtual parallel plane (VPP) networks and a logical traffic distribution from access points to the VPP networks.").
[117] *See* Munter, col. 4, lines 23-26; and Fig. 3.
[118] Munter, col. 6. Lines 18-36 (emphasis added).

Thus, as packets enter the network at the edge nodes, each POP switch distributes the packets across all of the parallel backbone networks (*i.e.*, across separate corresponding core switches within a core node). Where node and link capacities differ (*e.g.*, if link S1R1 and the corresponding link on S2, S2R1, have different capacities), the algorithm used to distribute traffic must "take into account" these differences.[119]

Munter further teaches that the POP switches distribute packets using a scheme that is "basically similar to what is frequently described as striping," wherein "**[a]ll packets** **sent** **from** **an edge node** 51-55 **towards** **the core nodes** 61-65 are systematically **distributed over** **all** **links**."[120] Thus, the load balancing taught by Munter is achieved by selectively interleaving (*i.e.*, "striping") packets, destined for a particular core node, across the exit ports of the POP switch that couple the POP switch to the destination core node switches. The interleaving balances traffic across core switches that are within different parallel planes of the destination core node. All of the load balancing taught by Munter is performed at the POP switches. No other load balancing is taught or suggested by Munter.

C. The Obviousness Rejections

4. Independent claims 1, 21, 41, 61, 81 and 101.

In rejecting the independent claims as allegedly obvious over Yamada in view of Munter, and further in view of Battle under 35 U.S.C. § 103(a), it was acknowledged in the Final Office Action that,

> Yamada does not teach at least one logical port having corresponded a plurality of physical ports to form a trunked group wherein frames in a trunked group are delivered in order; balancing frame traffic through the switch using the plurality of logical ports, with any frames exiting the switch via physical ports forming a trunked group being balanced over the physical ports forming the trunked group; and all ports operate at the same rate,[121]

but subsequently alleged that,

---

[119] Appellants note that Munter never teaches or suggests exactly *how* these differences are taken into account.
[120] Munter, col. 9, lines 60-63 (emphasis added).
[121] Final Office Action, ¶ 2, p. 3.

> Munter teaches at least one logical port having corresponded a plurality of physical ports to form a trunked group, wherein the corresponded physical port can be any of the plurality of physical ports exiting the switch **[Col. 5, lines 20-31]** wherein frames in a trunked group are delivered in order **[Col. 6, lines 24-29]**; balancing frame traffic through the switch using the plurality of logical ports **[Col. 5, lines 32-34]**, with any frames exiting the switch via physical ports forming the trunked group being balanced over the physical ports forming the trunked group **[Col. 6, lines 24-29]**.[122]

and further that,

> Munter further teaches the corresponded physical port can be any of the plurality of physical ports exiting the switch. Fig. 4, **40** shows connecting any of the physical ports which form the logical port as described in Col. 5, lines 20-28.[123]

Appellants respectfully traverse the characterization of the cited art, noting that, as will be explained in more detail below,

- The load balancing referenced in col. 5, lines 32-34 is the load balancing taught in col. 6, lines 24-29 and does not involve any type of load balancing across the "bundle" of "wavelength links" within the "logical port link" described in col. 5, lines 32-34.
- The traffic balancing and packet flow order preservation that is taught in col. 6, lines 24-29 is performed by directing packets to specific exit ports within the POP switches, not directing packets to specific exit ports within either the core switches or the optical cross connect
- The traffic balancing and packet flow order preservation that is taught in col. 6, lines 24-29 is implemented using a simple striping algorithm that distributes traffic across different core switches within separate parallel backbone networks, not using a two-tiered distribution of traffic that includes balancing traffic between ports within a single core switch.

---

[122] Final Office Action, ¶ 2, pp. 3-4 (bolding in original).
[123] Final Office Action, Response to Arguments, p. 2 (emphasis added); Appellants note that element 40 is the cross connect.

Based on the above, Appellants respectfully submit that,

- Munter does not teach balancing traffic across physical ports within a trunked group.
- Munter does not teach balancing traffic across logical ports that are within the same switch as the physical ports of the trunked group.
- Munter does not teach the two-tiered balancing required by the independent claims.

a. The Load Balancing of the Independent Claims

Appellants note, using independent claim 1 as a representative example, that the claimed method requires,

> **balancing frame traffic through said switch using said plurality of logical ports**, said frame traffic including frames exiting said switch via said physical ports, a selected physical port for at least one of said frames exiting said switch being selected based at least in part on said correspondence, **with any frames exiting said switch via physical ports forming a trunked group being balanced over said physical ports forming the trunked group**.

Thus, frames are first balanced across a plurality of logical ports, wherein for each logical port the part of the traffic so balanced that is directed to a logical port is routed to a physical port that is selected at least in part based upon its correspondence to one of the logical ports. Once the physical port is selected for sending the logical port's apportioned traffic, if the physical port is part of a trunked group *additional* balancing of the traffic *apportioned* to the logical port is performed across the physical ports that form the trunked group. The method therefore requires two separate levels of balancing (*i.e.*, a two-tiered balancing). At the first level, traffic is balanced across logical ports, and at the second level traffic apportioned to a logical port is subsequently balanced across the physical ports of a trunked group associated with the logical port.

b. Munter does not Teach Balancing across Trunked Ports

In rejecting independent claim 1 as allegedly obvious over the cited art, it was alleged in the Final Office Action that col. 6, lines 24-29 of Munter (which as noted above describes how the striping algorithm of the POP switches balances traffic across the parallel plane of a core node) teaches balancing traffic across physical ports forming the trunked group.[124] This was further elaborated upon in the Advisory Action of December 9, 2008 (hereinafter "Advisory Action"), where it was alleged that, "Munter teaches traffic is balanced to access all N parallel networks. This accessing of networks is obviously done using physical ports since any traffic going into network has to go through physical ports."[125] Appellants note that what is required by independent claim 1 is that the traffic be balance over physical ports within the switch that belong to a trunked group, not just *any* physical port as apparently inferred in the Advisory Action. Appellants respectfully submit that simply noting the presence of physical ports within the POP switches fails to teach all of the requirements of the claim element.

Appellants further note that it was also alleged in the Final Office Action that col. 5, lines 20-31 (which as noted above describes the port links that connect the core switches to the POP switches via the cross connect) teaches a logical port that is corresponded to a plurality of physical ports to form a trunked group.[126] But the port links described in the cited section from Munter (and shown Fig. 4) connect distinct ports that are within either the core switch or the cross connect, not distinct ports within the POP switches that are each corresponded with a logical port link. Also, even though the cited text describes "logical port links" as collections of "wavelength links," Munter makes no reference to any "logical ports," and such logical ports (together with any physical port that might be associated with each "wavelength link") at best might be *inferred* to exist on either end of the "logical port link," *i.e.*, within the core switch, and within the cross connect. Appellants thus respectfully submit that because the inferred physical ports that are alleged to be analogous to the claimed trunked ports are not within the POP switch, and because the balancing taught by Munter is performed within the POP switch, Munter does

[124] *See* Final Office Action, ¶ 2, p. 4.
[125] Advisory Action, p. 2.
[126] *See* Final Office Action, ¶ 2, pp. 3-4.

not teach or suggest that the balancing performed by the POP switch is applied across trunked physical ports.

Appellants additionally note that as previously noted, the load balancing taught by Munter is performed **across** core switches within a node, and thus across the ports within each POP switch connects it to each parallel plane switch. Munter does not teach or even suggest any type of load balancing across links (or ports) associated with a single core switch, such as the "bundle" of "wavelength links" associated with a "logical port link" (*e.g.*, port link S1R1 configured as such a "bundle"). All that matters in achieving the balancing taught by Munter is whether traffic has been balanced across each link (or group of links) associated with a different switch within a given core node. Only differences in the overall capacity of each such links are considered in the load balancing performed by the POP switches, and no mention is ever made as to how traffic distribution differences *within* such "logical port links" (*i.e.*, between "wavelength links") affect or alter such balancing.[127] Appellants thus respectfully submit that regardless of whether it is proper to analogize the inferred physical ports allegedly associated with the "wavelength links" to the trunked physical ports required by the claims, the balancing taught by Munter does not apply to such links since they only provide a path through a single switch within a single parallel plane, not across multiple planes across which the striping algorithm taught by Munter can balance traffic.

Therefore, for at least all of these reasons, Appellants respectfully submit that Munter does not teach or even suggest balancing frames over physical ports that form a trunked group and that are corresponded to a logical port, as required by independent claim 1.

[127] Appellants further note that if Munter is applied to the independent claims as proposed in the Final Office Action, no differences that can be taken into account exist between individual "wavelength links" within a logical port link, since the ports associated with said links were analogized in the Final Office Action to physical ports within a trunk, and these physical ports are expressly required by the independent claims to operate at the same rate.

c. Munter does **not** Teach Balancing across Logical Ports within a Switch that Includes the Trunked Group Ports

In rejecting independent claim 1 as allegedly obvious over the cited art, it was alleged in the Final Office Action that col. 5, lines 32-34 of Munter (which as noted teaches that port links can have different capacities if the load balancing algorithm accounts for the capacity imbalance) teaches balancing traffic through the switch using the plurality of logical ports.[128] As already explained above, Munter only teaches a load balancing algorithm that is applied to ports within the POP switches. Appellants note that because the only ports that may reasonably be inferred to be "logical ports" are the ports at either end of a "logical port link" (*i.e.*, either ports within a core switch or ports within the cross connect), the load balancing taught by Munter is not performed across logical ports that are defined within each POP switch.

Further, the cited text from Munter clearly states that all that the load balancing algorithm does is take into account the un-symmetry or imbalance caused by difference in capacity between port links. Appellants are at a loss to understand how the plain meaning of the sentence, "The [logical port] links may have different capacity as long as the un-symmetry, or the imbalance is taken into account,"[129] and the sentence, "This is done by a load balancing algorithm, as known in the art,"[130] could possibly be interpreted to mean that the striping algorithm performed at the POP switch balances traffic uses a plurality of logical ports. Appellants respectfully submit that even if it were reasonable to analogize the inferred logical ports to the logical port of independent claim 1, Munter merely teaches that **only the <u>capacity differences</u> of the logical port links** (**<u>if</u>** they exist) are used by the striping algorithm, **<u>not</u> the actual logical ports** associated with each logical port link.

Additionally, because (as already explained) the physical ports alleged in the Final Office Action to be analogous to the trunked physical ports required by the claims (*i.e.*, the ports in the core switch and cross connect for each "wavelength link") are not located within the POP switches, the frame balancing performed by the POP switch is not performed over **<u>both</u>** logical

---

[128] *See* Office Action, ¶ 2, p. 4.
[129] Munter, col. 5, lines 32-33.
[130] Munter, col. 5, lines 33-34.

ports **and** trunked physical ports that are **all** within the **same** switch, as required by independent claim 1. This would be true even if it were reasonable to analogize the inferred logical ports to the logical port required by independent claim 1.

Therefore, for at least these reasons, Appellants respectfully submit that Munter does not teach or even suggest balancing frame traffic through the claimed switch using logical ports defined within the switch, wherein the switch also includes the physical ports that form a trunked group that is corresponded to at least one logical port.

d. Munter does not Teach Two-Tied Balancing within a Single Switch

In rejecting independent claim 1 as allegedly obvious over the cited art, it was alleged in the Final Office Action that col. 5, lines 32-34 teaches balancing traffic through the switch using the plurality of logical ports, and that col. 6, lines 24-29 teaches that frames exiting the switch physical ports forming a trunked group are balanced over the physical ports forming the trunked group.[131] It was further reiterated in the Advisory Action that, "Munter teaches in Col. 5, lines 29-34 balancing over logical ports while in Col. 6, lines 25-29 Munter teaches balancing using physical ports."[132] Appellants note that as already explained above, Munter teaches selectively interleaving (*i.e.*, "striping") packets, destined for a particular core node, across the exit ports of the POP switch that couple the POP switch to the destination core node switches. The interleaving balances traffic across core switches that are within different parallel planes of the destination core node.

By contrast, as also already explained above, independent claim 1 requires first balancing traffic across a plurality of logical links, and then balancing any traffic apportioned by the first balancing to a particular logical link across multiple physical links if the physical links corresponded to the particular logical link form a trunked group. Thus the balancing required by independent claim 1 is implemented in two tiers, wherein the second tier of balancing (across the physical ports of the trunked group) is performed on an apportioned subset of the frame traffic that results from the first tier balancing (across the logical ports). Thus, even if for the sake of

---

[131] *See* Final Office Action, ¶ 2, p. 4.
[132] Advisory Action, p. 2.

argument it were possible to analogize the load balancing taught by Munter (which seeks to achieve balance between core switches in different parallel planes) to the frame balancing required by one of the elements of independent claim 1 (*e.g.*, to the initial load balancing between logical ports), there is no other load balancing taught by Munter (*i.e.*, a distribution of traffic that seeks a balance **other** than the balance between planes) which can be analogized to the separate and distinct balancing required by the other remaining element of independent claim 1 (*e.g.*, the load balancing between the physical ports of a trunked group).

Further, the load balancing across physical ports within a trunked group required by independent claim 1 is applied to a subset of frame traffic present on a single logical port that results from the load balancing applied to the overall frame traffic distributed across the logical ports. Thus, even if it were reasonable to somehow interpret Munter as applying the striping algorithm to both logical ports and physical ports with a logical port, this would be insufficient, since the algorithm taught by Munter simply interleaves packets across all ports, with the goal of achieving a single balance: a balance between parallel planes. Such a reading of Munter fails to take into account both the claim requirement that there are two separate and distinct balancing criteria, and the claim requirement that the second distribution of frames is separately performed on a subset of the first distribution of frames. Munter simply does not teach or even suggest either of these claim requirements.

For at least these reasons, Appellants respectfully submit that Munter does not teach or even suggest balancing frame traffic using a plurality of logical ports, and then subsequently balancing a subset of the frame traffic (previously apportioned to a logical port by the first balancing) over physical ports that form a trunked group corresponded to the logical port.

For at least all of the above-described reasons, Appellants respectfully submit that Munter does not teach or even suggest all of the limitations of independent claim 1, nor all of the limitations of any of the other independent claim, which include similar limitations as those of claim 1 and were rejected on similar grounds in the Final Office Action.[133] Further none of the other art cited, either alone or together overcomes the deficiencies of Munter. Appellants

---

[133] *See* Final Office Action, ¶ 2, pp. 5-7 (independent claim 21), pp. 7-8 (independent claim 41), pp. 8-9 (independent claim 61), pp. 10-11 (independent claim 81), and pp. 11-12 (independent claim 101).

therefore respectfully submit that none of the independent claims are rendered obvious by the cited art under 35 U.S.C. § 103(a).

5. Dependent claims 2, 6-20, 22, 26-40, 42, 46-60, 66-80, 82, 86-100 and 103-105

Because each of dependent claim 2, 6-20, 22, 26-40, 42, 46-60, 66-80, 82, 86-100 and 103-105 includes all of the limitations of their respective independent claims, Appellants respectfully submit that these dependent claims are also not rendered obvious over the cited art under 35 U.S.C. § 103(a) for at least the same reasons as those presented above with respect to the independent claims.

Appellants further note, with regard to the rejection of dependent claim 17, that it was stated in the Final Office Action that,

> Yamada further teaches a selected physical port is selected based on a source tag and/or destination tag added to the frame after the frame enters switch **[Col 8, lines 8-13]**.[134]

Appellants respectfully traverse the rejection, noting that it fails to take into account all of the elements of dependent claim 17. Appellants further note that, in response to Appellants previously submitted arguments, it was stated in the Final Office Action that,

> Applicant argues that Yamada does not teach selecting a physical port based on a tag added to the frame after the frame enters the switch. However, examiner disagrees. Yamada clearly teaches adding the tag after the frame enters the switch and selecting a physical port based on the tag. Applicant argues that Yamada makes a selection of physical port before the tag is added, the opposite of the claim requirement where the selection is based on the tag. However, examiner disagrees. The claim requires a selection of physical port based on tag added to the frame after the frame enters the switch. Yamada teaches applying a label and transmitting through appropriate physical port based on the label [See Fig. 3]."[135]

Appellants respectfully traverse the characterization of Appellants' previously submitted arguments, noting that Appellants did **<u>not</u>** argue that Yamada makes a selection of a physical port before the tag is added. Appellants if fact argued that Yamada only teaches a single

---

[134] Final Office Action, ¶ 2, p. 5 (bolding in original).
[135] Final Office Action, Response to Arguments, p. 2.

selection of a physical port, whereas dependent claim 17 requires **two successive port selections**, wherein the source and/or destination tag is added before the **second** selection, which is based at least in part upon the added tag.

Appellants note that dependent claim 17 requires "wherein a **selected** physical port for at least one of said frames exiting said switch is **further selected** based at least in part on a source tag and/or a destination tag added to said frame after said frame enters said switch" (emphasis added). Claim 17 clearly distinguishes between two separate selections by referring to a select**ed** physical port (past tense), and by requiring that the port be **further** selected. Appellants respectfully submit that the plain language of the claim clearly requires that a second selection be performed on a physical port that has already been selected in some manner. Further, claim 17 requires that the second selection be based (at least in part) upon the added tag, thus implicitly requiring that the addition of the tag be performed **before the second selection**. Because Yamada only teaches a single selection of a physical port based on the contents of the L1 mapping table shown in Fig. 6 of Yamada, Appellants respectfully submit that Yamada does not teach or suggest the second subsequent selection required by dependent claim 17.

For at least these reasons, and in addition to the reasons already presented, Appellants respectfully submit that none of the cited art, either alone or together, teaches or suggest all of the limitations of dependent claim 17, and thus does not render the claim obvious under 35 U.S.C. § 013(a). Additionally, because dependent claim 97 includes limitations similar to those of claim 17 and dependent claims 19 and 99 respectively depend up claims 17 and 97 (all of which were rejected on similar grounds[136]), Appellants also respectfully submit that dependent claims 19, 97 and 99 are also not rendered obvious under 35 U.S.C. § 103(a) by the cited art for at least the same reasons as presented with regard to dependent claim 17.

D. Conclusion

For the reasons stated above, Appellants respectfully submit that the Examiner erred in rejecting claims 1-2, 6-22, 26-42, 46-62, 66-82, 86-101 and 103-105, and respectfully request

---

[136] *See* Final Office Action, ¶ 2, p. 5.

reversal of the rejections of these claims. Appellants believe that no extensions of time or fees are required, beyond those that may otherwise be provided in documents accompanying this response. Nonetheless, in the event that additional extensions of time are necessary to allow consideration of this paper, such extensions are hereby petitioned under 37 CFR § 1.136(a), and any fees required (including fees for net addition of claims) are hereby authorized to be charged to Wong Cabello's Deposit Account No. 50-1922, referencing docket number 112-0134US.

Respectfully submitted,

March 2, 2009

/Roberto de León

Roberto de León, Reg. No. 58,967
Wong, Cabello, Lutsch,
Rutherford & Brucculeri, L.L.P.
20333 SH 249, Suite 600
Houston, TX 77070
(832) 446-2461 (direct line)
(832) 446-2424 (facsimile)
wcpatent@counselip.com

Filed Electronically

VIII. CLAIMS APPENDIX

1. (Previously Presented) A method of routing a flow of frames comprising:

applying a correspondence between a plurality of logical ports and a plurality of physical ports of a switch, at least one logical port having corresponded a plurality of physical ports to form a trunked group, wherein said corresponded physical ports can be any of said plurality of physical ports exiting said switch, wherein said corresponded ports all operate at the same rate and wherein frames in a trunked group are delivered in order; and

balancing frame traffic through said switch using said plurality of logical ports, said frame traffic including frames exiting said switch via said physical ports, a selected physical port for at least one of said frames exiting said switch being selected based at least in part on said correspondence, with any frames exiting said switch via physical ports forming a trunked group being balanced over said physical ports forming the trunked group.

2. (Original) The method of claim 1, wherein a selected physical port for each of said frames exiting said switch is selected based at least in part on said correspondence.

3.-5. (Cancelled)

6. (Previously Presented) The method of claim 1, wherein said balancing comprises applying a pseudo-random process to select a particular logical port as an egress port, said particular logical port being selected for a particular frame of said frames exiting said switch.

7. (Original) The method of claim 6, wherein applying a pseudo-random process comprises applying a hash function.

8. (Previously Presented) The method of claim 6, wherein said correspondence is employed to determine the physical port to which to route said particular frame based at least in part on the logical port selected as said particular logical port.

9. (Previously Presented) The method of claim 1, wherein said balancing comprises applying a pseudo-random process to select a particular logical port as an egress port, said particular logical port being selected for each particular frame of said frames exiting said switch.

10. (Original) The method of claim 9, wherein applying a pseudo-random process comprises applying a hash function.

11. (Previously Presented) The method of claim 9, wherein said correspondence is employed to determine the physical port to which to route each of said particular frames based at least in part on the logical port selected for each of said particular logical ports.

12. (Previously Presented) The method of claim 1, wherein said balancing comprises applying weights to select a particular logical port of said switch as an egress port for a particular frame of said frames exiting said switch.

13. (Original) The method of claim 12, wherein said correspondence is employed to determine the physical port to which to route said particular frame based at least in part on the selected logical port.

14. (Original) The method of claim 1, wherein a selected physical port for at least one of said frames exiting said switch comprises, for each of said frames exiting said switch, a physical port being selected based at least in part on said correspondence.

15. (Previously Presented) The method of claim 14, wherein said balancing comprises applying weights to select a particular logical port of said switch as an egress port for each of said frames exiting said switch.

16. (Previously Presented) The method of claim 15, wherein said correspondence is employed to determine the physical port corresponding to each of said particular logical ports to

which to route each of said frames exiting said switch, said corresponding physical port being based at least in part on the logical port selected as the particular logical port.

17. (Original) The method of claim 1, wherein a selected physical port for at least one of said frames exiting said switch is further selected based at least in part on a source tag and/or a destination tag added to said frame after said frame enters said switch.

18. (Original) The method of claim 17, wherein said source tag and/or said destination tag is stripped off said frame before said frame exits said switch.

19. (Original) The method of claim 1, wherein a selected physical port for each of said frames exiting said switch is further selected based at least in part on a source tag and/or a destination tag added to each of said frames after said frames enter, said switch.

20. (Original) The method of claim 19, wherein said source tag and/or said destination tag is stripped off each of said frames before each of said frames exits said switch.

21. (Previously Presented) An apparatus comprising:

a switch, said switch including a processor and memory;

said switch further including a plurality of logical and a plurality of physical ports, and having the capability to route a flow of frames exiting said switch;

said switch being adapted to apply a correspondence between said plurality of logical ports and said plurality of physical ports of said switch, at least one logical port having corresponded a plurality of physical ports to form a trunked group, wherein said corresponded physical ports can be any of said plurality of physical ports exiting said switch, wherein said corresponded ports all operate at the same rate and wherein frames in a trunked group are delivered in order, and being further adapted to balance frame traffic through said switch using said plurality of logical ports, said frame traffic including frames exiting said switch via said physical ports, a selected physical port for at least one of said frames exiting said switch being

selected based at least in part on said correspondence, with any frames exiting said switch via physical ports forming a trunked group being balanced over said physical ports forming the trunked group.

22. (Original) The apparatus of claim 21, wherein said switch is adapted to select a physical port for each of said frames exiting said switch based at least in part on said correspondence.

23.-25. (Cancelled)

26. (Previously Presented) The apparatus of claim 21, wherein said switch is adapted to balance frame traffic by applying a pseudo-random process to select a particular logical port as an egress port, said particular logical port being selected for a particular frame of said frames exiting said switch.

27. (Original) The apparatus of claim 26, wherein said switch is adapted to apply a pseudo-random process that comprises applying a hash function.

28. (Original) The apparatus of claim 26, wherein said switch is adapted to apply said correspondence to determine the physical port to which to route said particular frame based at least in part on the logical port selected as said particular port.

29. (Previously Presented) The apparatus of claim 21, wherein said switch is adapted to apply a pseudo-random process to select a particular logical port as an egress port, said particular logical port being selected for each particular frame of said frames exiting said switch.

30. (Original) The apparatus of claim 29, wherein said switch is adapted to apply a pseudo-random process that comprises applying a hash function.

31. (Previously Presented) The apparatus of claim 29, wherein said switch is adapted to employ said correspondence to determine the physical port to which to route each of said particular frames based at least in part on the logical port selected for each of said particular logical ports.

32. (Previously Presented) The apparatus of claim 21, wherein said switch is adapted to apply weights to select a particular logical port of said switch as an egress port for a particular frame of said frames exiting said switch.

33. (Original) The apparatus of claim 32, wherein said switch is adapted to employ said correspondence to determine the physical port to which to route said particular frame based at least in part on the selected logical port.

34. (Original) The apparatus of claim 21, wherein said switch is adapted to select a physical port for at least one of said frames exiting said switch, said physical port being selected based at least in part on said correspondence.

35. (Previously Presented) The apparatus of claim 34, wherein said switch is adapted to apply weights to select a particular logical port of said switch as an egress port for each of said frames exiting said switch.

36. (Previously Presented) The apparatus of claim 35, wherein said switch is adapted to employ said correspondence to determine the physical port corresponding to each of said particular logical ports to which to route each of said frames exiting said switch, said corresponding physical port being based at least in part on the logical port selected as the particular logical port.

37. (Original) The apparatus of claim 21, wherein said switch is adapted to selected a physical port for at least one of said frames exiting said switch based at least in part on a source tag and/or a destination tag added to said frame after said frame enters said switch.

38. (Original) The apparatus of claim 37, wherein said switch is adapted to strip said source tag and/or said destination tag off said frame before said frame exits said switch.

39. (Original) The apparatus of claim 21, wherein said switch is adapted to select a physical port for each of said frames exiting said switch based at least in part on a source tag and/or a destination tag added to each of said frames after said frames enter said switch.

40. (Original) The apparatus of claim 39, wherein said switch is adapted to strip said source tag and/or said destination tag off each of said frames before each of said frames exits said switch.

41. (Previously Presented) A switch fabric comprising:

at least a first switch and a second switch;

said first switch including a processor and memory;

said first switch further including a plurality of logical ports and a plurality of physical ports, and having the capability to route a flow of frames exiting said first switch;

said first switch being adapted to apply a correspondence between said plurality of logical ports and said plurality of physical ports, at least one logical port having corresponded a plurality of physical ports to form a trunked group, wherein said corresponded physical ports can be any of said plurality of physical ports exiting said switch, wherein said corresponded ports all operate at the same rate and wherein frames in a trunked group are delivered in order, and being further adapted to balance frame traffic through said first switch using said plurality of logical ports, said frame traffic including frames exiting said first switch via said physical ports, a selected physical port for at least one of said frames exiting said first switch being selected based

at least in part on said correspondence, with any frames exiting said switch via physical ports forming a trunked group being balanced over said physical ports forming the trunked group.

42. (Original) The switch fabric of claim 41, wherein said first switch is adapted to select a physical port for each of said frames exiting said first switch based at least in part on said correspondence.

43.-45. (Cancelled)

46. (Previously Presented) The switch fabric of claim 41, wherein said first switch is adapted to balance frame traffic by applying a pseudo-random process to select a particular logical port as an egress port, said particular logical port being selected for a particular frame of said frames exiting said first switch.

47. (Original) The switch fabric of claim 46, wherein said first switch is adapted to apply a pseudo-random process that comprises applying a hash function.

48. (Original) The switch fabric of claim 46, wherein said first switch is adapted to apply said correspondence to determine the physical port to which to route said particular frame based at least in part on the logical port selected as said particular port.

49. (Previously Presented) The switch fabric of claim 41, wherein said first switch is adapted to apply a pseudo-random process to select a particular logical port as an egress port, said particular logical port being selected for each particular frame of said frames exiting said first switch.

50. (Original) The switch fabric of claim 49, wherein said first switch is adapted to apply a pseudo-random process that comprises applying a hash function.

51. (Previously Presented) The switch fabric of claim 49, wherein said first switch is adapted to employ said correspondence to determine the physical port to which to route each of said particular frames based at least in part on the logical port selected for each of said particular logical ports.

52. (Previously Presented) The switch fabric of claim 51, wherein said first switch is adapted to apply weights to select a particular logical port of said first switch as an egress port for a particular frame of said frames exiting said first switch.

53. (Original) The switch fabric of claim 52, wherein said first switch is adapted to employ said correspondence to determine the physical port to which to route said particular frame based at least in part on the selected logical port.

54. (Original) The switch fabric of claim 41, wherein said first switch is adapted to select a physical port for at least one of said frames exiting said first switch, said physical port being selected based at least in part on said correspondence.

55. (Previously Presented) The switch fabric of claim 54, wherein said first switch is adapted to apply weights to select a particular logical port of said first switch as an egress port for each of said frames exiting said first switch.

56. (Previously Presented) The switch fabric of claim 55, wherein said first switch is adapted to employ said correspondence to determine the physical port corresponding to each of said particular logical ports to which to route each of said frames exiting said first switch, said corresponding physical port being based at least in part on the logical port selected as the particular logical port.

57. (Original) The switch fabric of claim 41, wherein said first switch is adapted to selected a physical port for at least one of said frames exiting said first switch based at least in

part on a source tag and/or a destination tag added to said frame after said frame enters said first switch.

58. (Original) The switch fabric of claim 57, wherein said first switch is adapted to strip said source tag and/or said destination tag off said frame before said frame exits said first switch.

59. (Original) The switch fabric of claim 41, wherein said first switch is adapted to selected a physical port for each of said frames exiting said first switch based at least in part on a source tag and/or a destination tag added to each of said frames after said frames enter said first switch.

60. (Original) The switch fabric of claim 59, wherein said first switch is adapted to strip said source tag and/or said destination tag off each of said frames before each of said frames exits said first switch.

61. (Previously Presented) A network comprising:

a host;

a physical storage unit;

a first switch and a second switch communicatively coupled to form a switch fabric;

said first switch and said second switch further communicatively coupled to said host and said physical storage unit;

said first switch including a processor and memory, and further including a plurality of logical ports and a plurality of physical ports;

said first switch being adapted to apply a correspondence between said plurality of logical ports and said plurality of physical ports, at least one logical port having corresponded a plurality of physical ports to form a trunked group, wherein said corresponded physical ports can be any of said plurality of physical ports exiting said switch, wherein said corresponded ports all operate at the same rate and wherein frames in a trunked group are delivered in order, and being further adapted to balance frame traffic through said first switch using said plurality of logical

ports, said frame traffic including frames exiting said first switch via said physical ports, a selected physical port for at least one of said frames exiting said first switch being selected based at least in part on said correspondence, with any frames exiting said switch via physical ports forming a trunked group being balanced over said physical ports forming the trunked group.

62. (Original) The network of claim 61, wherein said first switch is adapted to select a physical port for each of said frames exiting said first switch based at least in part on said correspondence.

63.-65. (Cancelled)

66. (Previously Presented) The network of claim 61, wherein said first switch is adapted to balance frame traffic by applying a pseudo-random process to select a particular logical port as an egress port, said particular logical port being selected for a particular frame of said frames exiting said first switch.

67. (Original) The network of claim 66, wherein said first switch is adapted to apply a pseudo-random process that comprises applying a hash function.

68. (Original) The network of claim 66, wherein said first switch is adapted to apply said correspondence to determine the physical port to which to route said particular frame based at least in part on the logical port selected as said particular port.

69. (Previously Presented) The network of claim 61, wherein said first switch is adapted to apply a pseudo-random process to select a particular logical port as an egress port, said particular logical port being selected for each particular frame of said frames exiting said first switch.

70. (Original) The network of claim 69, wherein said first switch is adapted to apply a pseudo-random process that comprises applying a hash function.

71. (Previously Presented) The network of claim 69, wherein said first switch is adapted to employ said correspondence to determine the physical port to which to route each of said particular frames based at least in part on the logical port selected for each of said particular logical ports.

72. (Previously Presented) The network of claim 61, wherein said first switch is adapted to apply weights to select a particular logical port of said first switch as an egress port for a particular frame of said frames exiting said first switch.

73. (Original) The network of claim 72, wherein said first switch is adapted to employ said correspondence to determine the physical port to which to route said particular frame based at least in part on the selected logical port.

74. (Original) The network of claim 61, wherein said first switch is adapted to select a physical port for at least one of said frames exiting said first switch, said physical port being selected based at least in part on said correspondence.

75. (Previously Presented) The network of claim 74, wherein said first switch is adapted to apply weights to select a particular logical port of said first switch as an egress port for each of said frames exiting said first switch.

76. (Previously Presented) The network of claim 75, wherein said first switch is adapted to employ said correspondence to determine the physical port corresponding to each of said particular logical ports to which to route each of said frames exiting said first switch, said corresponding physical port being based at least in part on the logical port selected as the particular logical port.

77. (Original) The network of claim 61, wherein said first switch is adapted to selected a physical port for at least one of said frames exiting said first switch based at least in part on a source tag and/or a destination tag added to said frame after said frame enters said first switch.

78. (Original) The network of claim 77, wherein said first switch is adapted to strip said source tag and/or said destination tag off said frame before said frame exits said first switch.

79. (Original) The network of claim 61, wherein said first switch is adapted to selected a physical port for each of said frames exiting said first switch based at least in part on a source tag and/or a destination tag added to each of said frames after said frames enter said first switch.

80. (Original) The network of claim 79, wherein said first switch is adapted to strip said source tag and/or said destination tag off each of said frames before each of said frames exits said first switch.

81. (Previously Presented) An article comprising: a storage medium having stored thereon instructions, that when executed, result in performance of a method of routing a flow of frames comprising:

applying a correspondence between a plurality of logical ports and a plurality of physical ports of a switch, at least one logical port having corresponded a plurality of physical ports to form a trunked group, wherein said corresponded physical ports can be any of said plurality of physical ports exiting said switch, wherein said corresponded ports all operate at the same rate and wherein frames in a trunked group are delivered in order;

balancing frame traffic through said switch using said plurality of logical ports, said frame traffic including frames exiting said switch via said physical ports, a selected physical port for at least one of said frames exiting said switch being selected based at least in part on said correspondence, with any frames exiting said switch via physical ports forming a trunked group being balanced over said physical ports forming the trunked group.

82. (Original) The article of claim 81, wherein said instructions, when further executed, result in: a selected physical port for each of said frames exiting said switch being selected based at least in part on said correspondence.

83.-85. (Cancelled)

86. (Previously Presented) The article of claim 81, wherein said instructions, when further executed, result in: said balancing comprising applying a pseudo-random process to select a particular logical port as an egress port, said particular logical port being selected for a particular frame of said frames exiting said switch.

87. (Original) The article of claim 86, wherein said instructions, when further executed, result in: said applying a pseudo-random process comprising applying a hash function.

88. (Previously Presented) The article of claim 86, wherein said instructions, when further executed, result in: said correspondence being employed to determine the physical port to which to route said particular frame based at least in part on the logical port selected as said particular logical port.

89. (Previously Presented) The article of claim 81, wherein said instructions, when further executed, result in: said balancing comprising applying a pseudo-random process to select a particular logical port as an egress port, said particular logical port being selected for each particular frame of said frames exiting said switch.

90. (Original) The article of claim 89, wherein said instructions, when further executed, result in: said applying a pseudo-random process comprising applying a hash function.

91. (Previously Presented) The article of claim 89, wherein said instructions, when further executed, result in: said correspondence being employed to determine the physical port to

which to route each of said particular frames based at least in part on the logical port selected for each of said particular logical ports.

92. (Previously Presented) The article of claim 91, wherein said instructions, when further executed, result in: said balancing comprising applying weights to select a particular logical port of said switch as an egress port for a particular frame of said frames exiting said switch.

93. (Original) The article of claim 92, wherein said instructions, when further executed, result in: said correspondence being employed to determine the physical port to which to route said particular frame based at least in part on the selected logical port.

94. (Original) The article of claim 81, wherein said instructions, when further executed, result in: a selected physical port for at least one of said frames exiting said switch comprising, for each of said frames exiting said switch, a physical port being selected based at least in part on said correspondence.

95. (Previously Presented) The article of claim 94, wherein said instructions, when further executed, result in: said balancing comprising applying weights to select a particular logical port of said switch as an egress port for each of said frames exiting said switch.

96. (Previously Presented) The article of claim 95, wherein said instructions, when further executed, result in: said correspondence being employed to determine the physical port corresponding to each of said particular logical ports to which to route each of said frames exiting said switch, said corresponding physical port being based at least in part on the logical port selected as the particular logical port.

97. (Original) The article of claim 81, wherein said instructions, when further executed, result in: a selected physical port for at least one of said frames exiting said switch being further

selected based at least in part on a source tag and/or a destination tag added to said frame after said frame enters said switch.

98. (Original) The article of claim 97, wherein said instructions, when further executed, result in: said source tag and/or said destination tag being stripped off said frame before said frame exits said switch.

99. (Original) The article of claim 81, wherein said instructions, when further executed, result in: a selected physical port for each of said frames exiting said switch being further selected based at least in part on a source tag and/or a destination tag added to each of said frames after said frames enter said switch.

100. (Original) The article of claim 99, wherein said instructions, when further executed, result in: said source tag and/or said destination tag being stripped off each of said frames before each of said frames exits said switch.

101. (Previously Presented) An article comprising: a storage medium having stored thereon instructions that, when executed, result in performance of a method of initializing a switch to route a flow of frames comprising:

initializing a correspondence between a plurality of logical ports and a plurality of physical ports of said switch, at least one logical port having corresponded a plurality of physical ports to form a trunked group, wherein said corresponded physical ports can be any of said plurality of physical ports exiting said switch, wherein said corresponded ports all operate at the same rate and wherein frames in a trunked group are delivered in order; and

further initializing said switch to balance frame traffic through said switch using said plurality of logical ports, said frame traffic including frames exiting said switch via said physical ports, a selected physical port for at least one of said frames exiting said switch being selected based at least in part on said correspondence, with any frames exiting said switch via physical

ports forming a trunked group being balanced over said physical ports forming the trunked group.

102. (Cancelled)

103. (Previously Presented) The article of claim 100, wherein said instructions, when further executed, result in: initializing said switch to balance frame traffic by applying a pseudo-random process to select a particular logical port as an egress port, said particular logical port being selected for a particular frame of said frames exiting said switch.

104. (Previously Presented) The article of claim 103, wherein said instructions, when further executed, result in: initializing said switch to apply a pseudo-random process to select a particular logical port as an egress port, said particular logical port being selected for each particular frame of said frames exiting said switch.

105. (Previously Presented) The article of claim 100, wherein said instructions, when further executed, result in: initializing said switch to balance frame traffic by applying weights to select a particular logical port of said switch as an egress port for a particular frame of said frames exiting said switch.

IX. EVIDENCE APPENDIX

None.

X. RELATED PROCEEDINGS APPENDIX

None.